wCorr Arguments

Paul Bailey

2016-03-04

This vignette explores two Boolean switches in the package. First, the `ML` switch allows for either a non-MLE (but consistent) esitimate of the nusiance parameters that define the binning process to be used (`ML=FALSE`) or for the nusiance parameters to be estimated using the MLE (`ML=TRUE`). Second the `fast` argument gives the option to use a pure R implementation (`fast=FALSE`) or an implementation that relies on the `Rcpp` and `RcppArmadillo` packages (`fast=TRUE`).

Numerical simulations show that the results are essentially unaffected by either of these switches and so it is recomended to use `fast=TRUE` and `ML=FALSE` which will drastically speed compuation.

The "wCorr Formulas" vignette describes the statistical properties of the correlation estimators in the package.

The `ML` switch

The correlation coefficients between two vectors of random variables that are jointly bivariate normal–call the vectors $\boldsymbol{X}$ and $\boldsymbol{Y}$.

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim N \left[\begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \Sigma \right]$$

where $N(\mathbf{A}, \mathbf{\Sigma})$ is the bivariate normal distribution with mean $\boldsymbol{A}$ and covariance $\mathbf{\Sigma}$.

Polyserial computation

the likelihood function for an individual observation of the polyserial is[1]

$$\Pr\left(\rho = r, \theta; Z = z_i, M = m_i\right) = \phi(z_i) \left[\Phi\left(\frac{\theta_{m_i+2} - r \cdot z_i}{\sqrt{1-r^2}} \right) - \Phi\left(\frac{\theta_{m_i+1} - r \cdot z_i}{\sqrt{1-r^2}} \right) \right]$$

The log-likelihood is then

$$\ell(\rho; z, m) = \sum_i w_i \ln\left[\Pr\left(\rho = r; Z = z_i, M = m_i\right)\right]$$

The derivatives of ℓ can be computed but are not readily computed and so when the `ML` argumet is set to `FALSE` (the default) a one dimensional optimization of ρ is calculated using `stats::optimize`. When the `ML` argument is set to `TRUE` a multi-dimensional optimization is done for ρ and $\boldsymbol{\theta}$ using `minqa::bobyqa`.

[1] See the "wCorr Formulas" vignette for a more complete description and motivation for the polyserial correlations's likelihood function.

Polychoric computation

the likelihood function for the polychoric is[2]

$$\Pr\left(\rho=r;P=p_i,M=m_i\right)=\int_{\theta'_{p_i+1}}^{\theta_{p_i+2}}dx\int_{\theta_{m_i+1}}^{\theta_{m_i+2}}dyf(x,y|\rho=r)$$

The log-likelihood is then

$$\ell(\rho;p,m)=\sum_i w_i\ln\left[\Pr\left(\rho=r;P=p_i,M=m_i\right)\right]$$

The derivatives of ℓ can be computed but are not readily computed and so when the `ML` argumet is set to `FALSE` (the default) a one dimensional optimization of ρ is calculated using `stats::optimize`. When the `ML` argument is set to `TRUE` a multi-dimensional optimization is done for ρ, $\boldsymbol{\theta}$, and $\boldsymbol{\theta}'$ using `minqa::bobyqa`.

General setup for the unweighted case

A simulation is run several times. For each itteration, the following procedure is used:

- select a value of n (the number of observations)
- select a true correlation coefficient ρ
- generate $\boldsymbol{X}$ and $\boldsymbol{Y}$
- select the value of t and t' (the number of bins for $\boldsymbol{M}$ and $\boldsymbol{P}$)
- select $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ (the bin boundaries for $\boldsymbol{M}$ and $\boldsymbol{P}$)
- confirm that at least 2 levels of $\boldsymbol{M}$ and $\boldsymbol{P}$ are occupied (if not, retrun to generating $\boldsymbol{X}$ and $\boldsymbol{Y}$)
- calculate and recording relevant statistics

ML switch

It is easy to prove the consistency of the $\boldsymbol{\theta}$ for the polyserial and $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ using the non-ML case. Similarly, for ρ, because it is an MLE that can be obtained by taking a derivative and setting it equal to zero, the results are asymtotically unbiased and obtain the Cramer-Rao lower bound.

This does not speak to the small sample properties of these correlation coefficients. Previous work has described their properties by simulation and so that tradition is continued below.

- plot that shows difference as a function of ρ and at $n=10$, $n=100, and $n=1000$ between `ML=FALSE` and `ML=TRUE` when `fast=TRUE`

fast switch

This section looks at the agreement between the pure R implementation of the optimizations and the `Rcpp` and `RcppArmadillo` impelemntation. The code can compute with either option by setting `fast=FALSE` (pure R) or `fast=TRUE` (Rcpp).

This is the summary of all differences between the `fast=TRUE` and `fast=FALSE runs` for the polyserial

- plot that shows difference as a function of ρ and at $n=10$ and $n=1000000$ between `fast=FALSE` and `fast=TRUE` when `ML=TRUE`

[2] See the “wCorr Formulas” vignette for a more complete description and motivation for the polychoric correlations’s likelihood function.

Implications for speed

The following plot shows the compute time versus speed.

- plot of compute time vs n for all four options. Each stop when the mean >= 20 seconds